IN THE SPECIFICATION:

Please replace the paragraph beginning at page 4, line 12, with the following rewritten paragraph:

--Figures 2A and 2B illustrate the use of minimum and maximum token buckets to control bandwidth provided to a queue. Referring to Figure 2A, a minimum token bucket [[200]] **200** contains 1560 tokens and a maximum token bucket [[202]] **202** contains 1565 tokens at a time when a sixty-four-byte packet [[204]] **204** arrives in an output queue with which the token buckets are associated. Referring to Figure 2B, when sixty-four-byte packet [[204]] **204** is transmitted as transmitted packet [[206]] **206**, sixty-four tokens are removed from each token bucket. In the illustrated example, minimum token bucket [[200]] **200** now has 1496 tokens and maximum token bucket [[202]] **202** now has 1501 tokens.--

Please replace the paragraph beginning at page 5, line 7, with the following rewritten paragraph:

--Since the bandwidth being provided to a queue depends on the number of tokens in the token bucket, it is desirable to control the token bucket refresh rate in order to provide the desired bandwidth. However, available literature on token buckets does not specify how to set a token bucket refresh rate in order to archive a desired bandwidth value. In addition, an end user of a switched network element may desire to specify bandwidth in standard ~~dominations~~ denominations, such as kilobits per second, rather than token bucket refresh rates. Current literature on token buckets likewise does not specify how to convert standard bandwidth denominations into token bucket refresh rates. For example, Perros, An Introduction to ATM Networks, John Wiley and Sons, (2001) states

that token buckets can be used for admission control in ATM networks. However, details on converting bandwidth values to token bucket refresh rates are not provided.--

Please replace the paragraph beginning at page 12, line 6, with the following rewritten paragraph:

--Figure 5 is a flow chart illustrating exemplary steps for fine grain bandwidth allocation according to an embodiment of the present invention. The steps illustrated in Figure 5 may be implemented by traffic manager **412** illustrated in Figure 4 based on bandwidth values input by user via user interface **310** and converted by bandwidth converter **314**. Referring to Figure 5, in step **500**, user interface **310** receives input from a user regarding bandwidth to be allocated to a queue. The user may specify bandwidth in any suitable standard bandwidth ~~domination~~ denomination, such as kilobits per second. The user may specify a single bandwidth value to be allocated to a queue or minimum and maximum bandwidth values. In step **502**, the bandwidth value is converted into a base bandwidth value and a residual bandwidth value. The base bandwidth value may be set to a predetermined value that may be guaranteed every token bucket refresh interval and a residual bandwidth value that may be spread over multiple token bucket refresh intervals. For example, the base bandwidth value may be on the order of one megabit per second and the residual bandwidth value may be on the order of less than one megabit per second. In one example, if a user desires to allocate 1.1 megabits per second to a queue, the base bandwidth value may be set to one megabit per second and the residual bandwidth value may be set to .1 megabits per second. Similarly, if the user desires to allocate 9.1 megabits per second to the queue, the base bandwidth value may be set to nine megabits per second and the residual bandwidth value may be set to .1 megabits per

second. By guaranteeing a base bandwidth value every token bucket refresh interval and providing the residual bandwidth value over multiple token bucket refresh intervals, the present invention allows the user to specify any increment of bandwidth to be allocated to a queue and to allocate such bandwidth without burstiness.--